**Заключение**

Если следовать концепции разработки веб-фреймворка Django и его документации, писать тщательно обдуманный и хорошо спроектированный код, то большая часть популярных уязвимостей в проектах, основанных на нем, не работает даже в базовой настройке. Однако для тех уязвимостей, которые не решает Django, существует множество сторонних компонентов, которые позволяют уберечься от атак подобного рода.

## ПРИМЕНЕНИЕ MIDI-ФОРМАТОВ В КАЧЕСТВЕ СТЕГАНОГРАФИЧЕСКИХ КОНТЕЙНЕРОВ

*С. А. Неймышева*
(Екатеринбург, УрГУПС)

Современный прогресс в области глобальных компьютерных сетей и средств мультимедиа привел к разработке новых методов, предназначенных для обеспечения безопасности передачи данных по каналам телекоммуникаций и использования их в необъявленных целях, и имеет большое значение для развития автоматизированного рабочего места (АРМ) сотрудника. В связи со всеобщей компьютеризацией появилась необходимость использования планшетных ПК в рабочих целях. Так как планшетные ПК позволяют работнику быть мобильным, то очень часто он находится вдали от рабочего места, тем самым ограничивается доступ к определенным рабочим ресурсам. К примеру, для получения одноразового пароля используют программы, не защищенные от перехвата пароля. Эту проблему могут решить методы стеганографии, которые скрывают сам факт передачи сообщения, используя стегоконтейнеры. Эти методы, учитывая естественные неточности устройств оцифровки и избыточность аналогового видео- или аудиосигнала, позволяют скрывать сообщения в компьютерных файлах (контейнерах).

В настоящее время методы компьютерной стеганографии развиваются по двум основным направлениям:

1. Методы, основанные на избыточности аудио- и визуальной информации. Младшие разряды цифровых отсчетов содержат очень мало полезной информации. Заполнение их дополнительными данными практически не влияет на качество восприятия.

2. Методы, основанные на использовании специальных свойств компьютерных форматов.

Среди них можно выделить следующие: методы использования зарезервированных для расширения полей компьютерных форматов данных, методы специального форматирования текстовых файлов, методы скрытия в неиспользуемых местах гибких дисков, методы использования имитирующих функций, методы удаления идентифицирующего файл заголовка [2].

Выбор и применение того или иного стеганографического метода полностью зависят от типа используемого контейнера и требований, предъявленных к процессу записи сообщения в контейнер и последующей передачи этого контейнера от отправителя к получателю.

Стандартный MIDI-файл – это специально разработанный формат файлов, предназначенный для хранения данных, записываемых и/или исполняемых секвенсером. Секвенсер может быть как программой для компьютера, так и аппаратно выполненным модулем.

MIDI (англ. Musical Instrument Digital Interface – цифровой интерфейс музыкальных инструментов) – стандарт цифровой звукозаписи на формат обмена данными между электронными музыкальными инструментами.

Интерфейс позволяет единообразно кодировать в цифровой форме такие данные, как нажатие клавиш, настройка громкости и другие акустические параметры, выбор тембра, темпа, тональности и др., с точной привязкой во времени. В системе кодировок присутствует множество свободных команд, которые производители, программисты и пользователи могут использовать по своему усмотрению. Поэтому интерфейс MIDI позволяет, помимо исполнения музыки, синхронизировать управление другим оборудованием, например, осветительным, пиротехническим и т. п. [4].

Последовательность MIDI-команд может быть записана на любой цифровой носитель в виде файла, передана по любым каналам

связи. Воспроизводящее устройство или программа называется синтезатором (секвенсором) MIDI и фактически является автоматическим музыкальным инструментом.

Файлы с оцифрованным звуком содержат значения амплитуды звукового сигнала, измеренные через одинаковые промежутки времени, в то время как файлы с нотной записью (в частности MIDI) – последовательность команд, сообщающих, какую ноту, каким инструментом и как долго нужно воспроизводить в тот или иной момент времени. В связи с этим количество информации в файлах с нотной записью в сотни и тысячи раз меньше, чем в файлах с оцифрованным звуком [5]. Это заметно ограничивает возможности записи данных методами, основанными на избыточности информации файла-контейнера. Неудивительно, что в этом направлении проводится гораздо меньше исследований. Однако использование специальных свойств форматов может предоставить большие возможности для скрытия информации от посторонних глаз.

В этом формате хранятся стандартные MIDI-сообщения (т. е. статус-байты и соответствующие им байты данных), а также временные метки или маркеры для каждого сообщения (т. е. последовательности байтов, указывающие, какое количество условных единиц времени необходимо подождать перед тем, как исполнить следующее событие MIDI) [3]. Этот формат позволяет сохранять информацию о темпе, временном разрешении, выраженном в количестве тиков на одну четвертную длительность, обозначении размера, информацию о музыкальных ключах, а также хранить названия треков и паттернов [1]. Формат предусматривает возможность сохранения в одном файле нескольких паттернов и треков таким образом, что программы-приложения могут выбирать из всего набора хранимой информации ту, которая будет понятна данному приложению.

Формат разработан так, чтобы любой секвенсер мог читать и записывать MIDI-файл таким образом, чтобы не потерялись его данные, и так, чтобы формат был достаточно гибким, т. е. чтобы приложения могли сохранять в файлах

Первый из предлагаемых методов скрытия информации в MIDI-файле основан на способности секвенсеров считывать информа-

цию только из записей известных им типов. Можно стандартизировать какой-либо из несуществующих типов записей и использовать его в качестве хранилища данных, причем при указании в начале такой записи ее размера абсолютно исключается возможность обнаружения «различий на слух», так как все секвенсеры ее будут пропускать.

Однако использование данного метода влечет за собой двоякое понимание требования «прозрачности». По аналогии с файлами других форматов это должно означать отсутствие (точнее, сведение к минимуму) отличий в восприятии незаполненного и заполненного файла-контейнера, другими словами, этого не должно быть слышно. Как было сказано выше, в этом варианте требование «прозрачности» выполняется. Но в отличие от секвенсеров любой специалист наверняка знает, какой набор байт может, а какой не может быть самостоятельной и «безобидной» записью, поэтому при непосредственном просмотре двоичного кода такая запись с высокой вероятностью будет обнаружена, что заметно снижает эффективность ее использования [1].

Следующий из предлагаемых методов также использует свойства формата MIDI. Предлагается записывать информацию в файл посредством генерирования нот определенной высоты и, возможно, длительности. С целью уменьшения вероятности обнаружения сообщения генерацию нот следует проводить по определенным законам существующих музыкальных гармоний. Так, например, можно расставлять ноты в соответствии с любым натуральным ладом, в данном случае в пределе одной октавы будем иметь семь различных нот. Во избежание возможных больших «скачков» в тональностях двух соседних нот следует ограничиться рассмотрением лишь названия ноты, не принимая во внимание октаву за одним исключением: чтобы один ключ (т. е. одна нота в MIDI-файле) нес целое число бит информации, необходимо добавить еще одну ноту, обособив ее от остальных. К примеру, можно каждую ноту «до» четной октавы отличать от ноты «до» нечетной октавы. В данном случае получается метод записи, абсолютно не заметный при визуальном просмотре кода, но тем не менее ощутимый на слух, при котором

каждый ключ MIDI-файла может нести в себе 3 бита дополнительных данных. Особо обратим внимание на то, что при этом методе передаваемая информация не просто записывается в контейнер, а сама впоследствии становится его частью, после чего файл с уже скрытой информацией может рассматриваться как пустой контейнер, в который можно записать другие данные, правда, уже иным методом. С другой стороны, в качестве контейнера можно рассматривать MIDI-файл без нот, в этом случае данный метод не является принципиально отличным, однако в такой пустой контейнер можно записать информацию далеко не всеми оставшимися методами.

Другими методами, использующими свойства формата MIDI, может служить запись данных в специальные поля, отведенные для текста композиции, названий треков, комментариев, информации об авторе, в значение темпа композиции, в переменные, отвечающие за расстановку нот по каналам, правда, при различной емкости этих элементов контейнера такие данные будут обнаружены с весьма высокой степенью вероятности, поэтому особое внимание на них заострять не будем.

Среди методов, основанных на избыточности информации в пустом контейнере, следует выделить запись данных в младшие биты временных меток (delta-time), а также статической характеристики каждой конкретной ноты – velocity (сила нажатия). Выделив по 3 младших бита в каждом элементе контейнера, можно получить весьма слабо заметные отличия заполненного контейнера от пустого как на слух, так и при визуальном исследовании двоичного кода файла (изменения в delta-time визуально определить практически невозможно). В пользу этих методов выступает возможность записывать непосредственно сыгранные музыкантом партии в виде команд в MIDI-файл. Очевидно, что как сила нажатия на клавишу, так и интервалы между нотами в данном случае никак не могут быть абсолютно одинаковыми, что уменьшает вероятность обнаружения скрытой информации практически до нуля.

Хотя цифровая стеганография является относительно новой областью исследований, развитие цифровых сред и связанные с этим практические потребности стимулируют интенсивный рост интере-

са к этой области во всем мире и чрезвычайно высокую активность исследователей в последнее десятилетие. К настоящему моменту в данной области предложено достаточно большое множество методов, технологий и инструментальных средств. Огромный интерес к стеганографии в значительной мере стимулируется быстрым распространением Интернета, возрастанием роли интернет-технологий и соответствующих цифровых сред в практической жизни общества. Последнее ведет к необходимости защиты аудио- и видеоинформации перед тем, как она будет публиковаться в Интернете или распространяться средствами Интернета [3]. Существует также много других причин, обусловливающих чрезвычайно высокий интерес к использованию методов стеганографии, в частности, в военном деле, юриспруденции, электронной коммерции, а также других областях, где необходимо обеспечивать секретность коммуникаций, защищать авторские права, предотвращать пиратское использование программного продукта и т. п.

Использование MIDI-форматов для стеганографических контейнеров позволяет скрывать меньший объем секретной информации, чем использование графических и аудиоформатов, тем не менее при огромной распространенности различных методов стеганографии в последних разработках алгоритмов для MIDI предоставляет весьма большие возможности, в первую очередь из-за кажущейся непригодности формата для данных целей, а следовательно, и гораздо более низкой степени контроля при проверке файлов на предмет содержания скрытых сообщений. Не следует упускать из виду и немалый потенциал MIDI-стеганографии как одного из наименее исследованных направлений современной компьютерной стеганографии.

## Библиографические ссылки

1. *Голубев В.* Компьютерная стеганография – защита информации или инструмент преступления? [Электронный ресурс]. Режим доступа: http://www.crime-research.ru/

2. *Городецкий В. И., Самойлов В. И.* Стеганография на основе цифровых изображений. СПб. : С.-Петербур. ин-т информатики и автоматизации РАН, 1986.

3. *Барсуков В. С., Романцов А. П.* Компьютерная стеганография вчера, сегодня, завтра. Технологии информационной безопасности XXI века // Специальная техника. 2008. № 4–5 [Электронный ресурс]. Режим доступа: http://st.ess.ru/

4. *Шнайер Б.* Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. М. : Изд-во ТРИУМФ, 2003.

5. *Новосельский А.* Форматы звуковых файлов // Компьютеры + Программы. 2006. № 1 [Электронный ресурс]. Режим доступа: http://www.aip.mk.ua/

## АНАЛИЗ ЗАЩИЩЕННОСТИ ОБЛАЧНЫХ СИСТЕМ С ПОМОЩЬЮ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

*Т. И. Паюсова*
(Тюмень, ТюмГУ, database_kb@mail.ru)

Научный руководитель: д-р техн. наук, профессор *А. А. Захаров*

Облачные системы являются принципиально новой концепцией предоставления сетевого доступа к вычислительным и компьютерным ресурсам, например, к хранилищу данных, серверу, приложению, виртуальной машине. Основными характеристиками облачных инфраструктур являются: объединение ресурсов в пул, широкая доступность через сеть, самообслуживание по требованию пользователя (заказчика), гибкость и масштабируемость, измеримость предоставляемых услуг.

Объединение ресурсов в пул является реализацией принципа множественной аренды (multi-tenancy): предоставляемые ресурсы объединяются в пул и динамически распределяются между большим числом пользователей в соответствии с их потребностями.

Широкая доступность облачных сервисов подразумевает, что доступ к ним можно получить с помощью стандартных средств и механизмов, используя разнородные клиентские платформы. Кроме этого, пользователь в случае необходимости может получить доступ к облачным вычислениям самостоятельно без взаимодействия с поставщиком.